Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 159 (2019) 590–599

Procedia Computer Science

www.elsevier.com/locate/procedia

23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Knowledge-Based Architecture for Recognising Activities of Older People

Mohamed Bennasar[1*], Blaine A. Price[1], Avelie Stuart[2], Daniel Gooch[1], Ciaran McCormick[1], Vikram Mehta[1], Linda Clare[2], Amel Bennaceur[1], Jessica Cohen[5], Arosha K. Bandara[1], Mark Levine[4] and Bashar Nuseibeh[1,3]

[1] *School of Computing and Communications, The Open University, UK*
[2] *Psychology, University of Exeter, UK*
[3] *University of Limerick, Ireland*
[4] *Lancaster University, UK*
[5] *Age UK Exeter, UK*

Abstract

The world is facing an ageing population phenomenon, coupled with health and social problems, which affect older people's ability to live independently. This situation challenges the viability of health and social services. Smart home technology can play a significant role in easing the pressure on caregivers, as well as reduce the financial costs of health and social services. Activity of Daily Living (ADL) recognition is an essential step to translate sensor data into activities at high semantic levels. Supervised Machine Learning (ML) algorithms are the most commonly used techniques for this application. However, a common problem is a lack of availability of enough annotated data to train these algorithms. Collecting annotated data is expensive, time consuming, and may violate people's privacy. Intra- and inter-personal variation in performing complex activities is another challenge for an ML-based activity recognition approach. In this paper, a multi-layered knowledge-based architecture for recognising ADL in real-time is proposed. At the first stage, sensor data is pre-processed; events that describe changes in the environment are detected at the second stage, in which the sequence of events is used to recognise more semantically complex activities at the third stage. A new ADL ontology is proposed to model the knowledge related to the sensor platform and the targeted activities as the previously proposed ontologies were either designed to deal with specific sensor data, or they ignored the context environment information which is important in recognising complex activities.

Keywords: Older People Care, Activity Recognition, ADL, Smart Home, Knowledge-based Architectures, Sensor Event Detection;

* Corresponding author. Tel.: +44 (0) 1908655019.
E-mail address: Mohamed.Bennasar@Open.ac.uk

1877-0509 © 2019 The Authors. Published by Elsevier B.V.

10.1016/j.procs.2019.09.214

1 Introduction

According to the United Nations, the world is facing an ageing population phenomenon, with the number of people aged 60 or older was 962 million in 2017, twice as high as the figures for 1980. This figure is expected to reach 2.1 billion in 2050. In fact, the number of older people is expected to exceed the number of children aged 10 years or under. Moreover, by 2050, the prediction model shows that the number of older people aged over 60 years will be greater than the number of people aged between 10-24 years [1]. Older people who live alone face risks that range from their safety, to their health and wellbeing [2]. As a result of this rapid demographic change, health and social services are facing huge pressure and must adopt cost effective care models.

Ambient Assisted Living (AAL) technology is currently gaining a great deal of attention, driven by its capability to satisfy the needs of older people and reduce the demands on social care and intervention [3]. AAL systems can be key to enabling the support of older people in living in their own homes, offering them cost effective care, as opposed to human care-based services [4].

The AAL system can employ a variety of sensors to gather information about the home's inside environment, along with information about the occupants who live there. This information provides deep insight about the occupants' ADL, as well as monitoring safety [5]. Sensors have been used for activity monitoring and recognition since the 90s [6]. A variety of sensors are used for activity recognition, such as wearables, environmental, audio, Passive Infra-Red (PIR), energy, and cameras. However, they can be classified into two main classes in terms of their placement and information captured.

First, wearable sensors-based activity recognition, which include accelerometers [7], GPS [8],and biosensors [9] can be attached to the human body to monitor physical activities, position, and vital signs. Second is dense sensing, which are sensors attached to objects and placed around the house to measure temperature, humidity, pressure, light, gas, door states, energy consumption, and occupancy. These are useful for detecting the interaction between the user and the environment [10]. In the context of older people's care, recognising the ADL of older people living at home can enhance their independence, help caregivers, and provide more information about life patterns and health states, as well as improve their sense of security and confidence [11].

Activity recognition based on sensor data is used often in the area of AAL. However, it has been applied to many other applications in a variety of fields, such as mobile computing [12], security [13], and context awareness [14]. ADL monitoring is also used with AAL for long-term health monitoring or instant assistance. ADL recognition is a multistage process that begins with choosing the appropriate sensors to capture activities. Second, the collected data is prepared for the appropriate abstraction level. Third, the data is pre-processed and transferred to a suitable format for the recognition algorithms. Finally, the computational recognition algorithms are developed to infer the activities [15].

In this paper, a novel knowledge-based activity recognition system is proposed, designed to detect events and recognise activities in a two-level manner in real-time to provide caregivers with information when it is needed. The proposed system is designed to recognise complex kitchen and bathroom activities. This architecture is proposed as part of the STRETCH (Socio-Technical Resilience for Enhancing Targeted Community Healthcare) project. The overall aim of the project is to develop a smart system for enhancing care quality, safety, and security for older people that can improve their ability to live independently in their own homes. The proposed architecture is designed to work with sensor data that is collected by the STRETCH sensor platform [16]. A knowledge-based approach is chosen because of the difficulty in collecting enough annotated data from older participants' homes.

The main contribution of this paper is the proposed knowledge-based architecture to overcome the lack of annotated data, which has been the main problem with using ML-based activity recognition. The proposed architecture is designed to be easily adopted for various smart environments and can overcome intra- and inter-personal variation problems. It is also easily modified; therefore, it can be deployed in ML algorithms to become a hybrid solution when annotated data becomes available.

The rest of this paper is structured as follows: Section 2 discusses the related work, Section 3 presents the proposed ontology, architecture, and rules of each layer in the architecture, and Section 4 concludes the paper with a discussion of planned future work.

2 Related work

Activity recognition can be classified into two approaches: Data-driven, and knowledge driven activity recognition. The data-driven approach relies on ML algorithms to discover and learn the patterns from the sensor data and perform recognition. Annotated data is usually needed to train ML algorithms before they can be used to perform recognition tasks. The main advantage of the data-driven approach is its ability to handle uncertainty, and some of these techniques are able to handle temporal information (e.g., a recurrent neural network). However, ML algorithms require a relatively large annotated datasets for capturing most of the patterns related to ADLs. Collecting such large datasets is expensive and time consuming; it may also be a privacy violation for the participants because it may involve entering their homes and watching their activities. The performance of the algorithms can be lower when the learned activity model from one participant's data is applied on another [17]. Moreover, each dataset has its own unique quality, type of sensors, sampling frequency, type of activities, and application, which can make reusing publicly available datasets very difficult [18].

A knowledge-based activity recognition approach uses prior knowledge about the activity sequence to produce a logic-based recognition model. Unlike a data-driven approach, knowledge-based techniques do not require large annotated datasets, and are also semantically clearer and easy to start. This approach can be used to overcome the scarcity of the annotated data. However, they are not robust in handling uncertainty [15]. The knowledge-based approach is driven by observational knowledge about ADL performance, objects, and devices involved, as well as a sequence of actions and events for each activity. People tend to perform activities in their own personal way. However, there is no significant difference in the involved objects and resources [15]. This knowledge-based approach relies on definitions of the involved components and semantics to model the complex activities [19].

The knowledge structure of these definitions is modelled and expressed through various formats, such as logical axioms, rules, and ontologies [20, 21]. The knowledge structure is modelled using various means, including the mining-based approach, which uses information retrieval techniques to retrieve definitions of activities and phrases that describe involved objects and the activity performance process. Several studies have been conducted using this approach [22, 23]. Another approach used is the logical-based approach, in which a variety of logical formalisms are used to represent the knowledge about the activity [24]. Finally, ontologies have been used to model the knowledge related to the mechanism of activities, objects, actors, and sensors. The term ontology is from philosophy and is used to deal with existence and things that exist. In the knowledge engineering field, it is defined as the "specification of a conceptualisation" [25]. Ontologies are also used to conceptualise the activities and their interrelationship. They are also used to model the contextual information to construct the information at any time [26, 27].

An ADL Ontology for Ubiquitous Systems has been proposed as part of SPHERE project [18], which is a hierarchical ontology for ADL within a smart home. This ontology has been developed specifically to represent the data collected by the SPHERE sensor platform which includes data from 3D video cameras. OBO format is used to develop this ontology which contains 165 different activities organised hierarchally. Activities are grouped into three groups: Health conditions, Social interaction, and atomic home activities. Salguero et al. [28], proposed an ontology for ADL representation. However, this ontology contains only three concepts: activity, event, and sensor. It also focused only on the sensor events and ignored the other context information such as location of activity, actors, resources, etc. Another ADL ontology has been proposed in [29], which was developed as part of an ADL monitoring system based on mobile networks. It consists of concepts and properties that are used to model message communication between user and the ADL system. Unfortunately, this ontology is not publicly available.

Reusing previously proposed ontologies is problematic, as some of them may be designed for a specific application and may not fit with the sensor platform data or may be very large, descriptive and computationally expensive to use [30].

3 Methods

The proposed ADL recognition architecture is designed to work on data collected by the STRETCH sensor platform. A high-level overview of STRETCH system architecture is described in Figure 1. Moreover, more details about the architecture can be found in [16]. The data is streamed in real-time to the STRETCH back-end server, where recognition takes place. This platform has been designed to monitor older people who live on their own homes. Therefore, it employs only non-invasive sensors, which are very efficient in power consumption and do not require much attention (long life batteries that do not require recharging and a system that is remotely administered).

The STRETCH platform employs three different types of sensing modalities to provide the information needed about the home environment and the older person to provide a generic ADL recognition. These include, an environmental sensor network, where each unit consists of humidity, light, temperature, pressure, and PIR sensors. An energy consumption sensor is also part of this platform, along with a wearable triaxial accelerometer that is attached to the participant's wrist to monitor body movements. The proposed ADL recognition architecture will be designed to deal also with information from sleep sensors in the bed and door sensors on the fridge and front door, in addition to STRETCH platform sensors.

3.1 Older People ADL Ontology

A list of ADLs was evolved from meetings with multidisciplinary professionals who are involved in the care of older people, which includes activities related to nutrition, in taking fluids, sleep, hygiene and showering. However, there is no a such ontology fit for the purpose of this study because either they were designed to deal with specific sensor data, or they ignored the context environment information that is important in recognising complex activities. An ontology (Fig 2 shows the main concepts) has been proposed, which considers both the environment context information and sensor data (image of complete ontology is available online: https://embed.kumu.io/2f5f0715309b0b28a89ef92e4c3a5287). It is merges dynamic information collected from STRETCH sensor platform, static information about the activities, and home environment. The ontology consists of seven main classes including: activity name, social context, activity level, location, resources, related event and duration. Each of the main seven classes has subclasses, which are discussed in the following sections. The concept of the entities are chosen to suit the dataset collecting by STRETCH sensor platform.

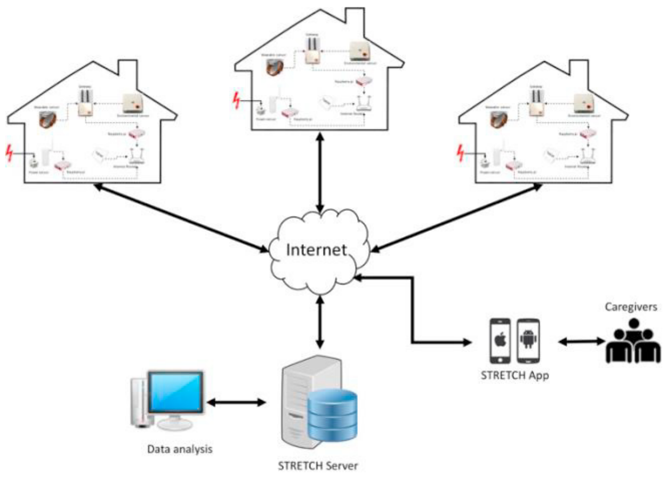

Fig. 1. An overview of STRETCH system architecture.

Four sets of activities are being considered in this ontology: activities performed in the kitchen, bedroom, bathroom, and living room, which include preparing a meal, preparing a drink, watching television, reading/resting, sleeping, showering, and other hygiene activities. These activities may involve a resource, which can be kitchen appliance, a bed, or a television. The activities are related to events that can change the environmental context, such as, the steam level increases, a door is opened, a device is switched on, a light is switched on, or a change in the participant's activity level. The activity can be performed by the older person or by a caregiver, so each activity has a social context.

3.2 STRETCH Knowledge-based System Architecture

The activity recognition is performed using a knowledge-based architecture, which consists of three levels. Figure 3 depicts the architecture. The STRETCH ontology is used to model the ADLs and the context of the smart environment. Using a knowledge-based approach is not only to avoid the scarcity of the annotated data to train the ML algorithms, but it is also used to combine the domain knowledge within the ADLs and context models. As shown in Fig 3, activity recognition is done in three layers.

The first layer is the raw data of humidity, PIR, light, wearables, bed, and energy sensors, which are pre-processed. Data collected by the sensor platform is segmented using an events-based approach. During the activity, the state of the sensors is changed based on the type of the activity. During showering for example, the reading of humidity sensor is expected to increase above a specific threshold, and the PIR sensor in the bathroom is expected to fire. The sampling frequency of the PIR sensor is 60 Hz which means it sends a reading every second. Therefore, the occupancy of each part of the house is updated every second. The rest of the sensors are event driven, they send a reading when there is change in the environment, these changes are used to detect the events. For the wearable sensor, the average acceleration will be calculated every second, which means the activity level is updated every second.

The second layer is called the event detector. Due to the limitation of detailed actions that the sensor platform can capture, the events are defined as manifestations. It employs a set of rules on sensor data to detect changes in the contextual environment. For example, if the humidity readings have changed above a certain threshold at time t, then the event at time t is steam_on. EV(Steam_on, t) is used to denote the related event. The third layer is the semantic rule-based reasoner, which performs activity recognition using the information about detected events. For example, the sequence of events: EV(steam_on, t1), EV(steam_off, t2), EV(kettle_on, t3) , where EV(kettle_on, t4) are strongly related with preparing a drink activity during time t3 - t4.

The architecture is proposed this way for two reasons; first, to be able to replace the semantic rule-based reasoner with ML techniques when enough annotated data becomes available (the ML will be trained using the events, rather than raw sensor data), and second, to use the information about event sequence and recognised activity to feed the layers of behavioural analysis and anomaly detection. More details about the rules of the events detector and semantic rule-based reasoner layers are explained in the following sections.

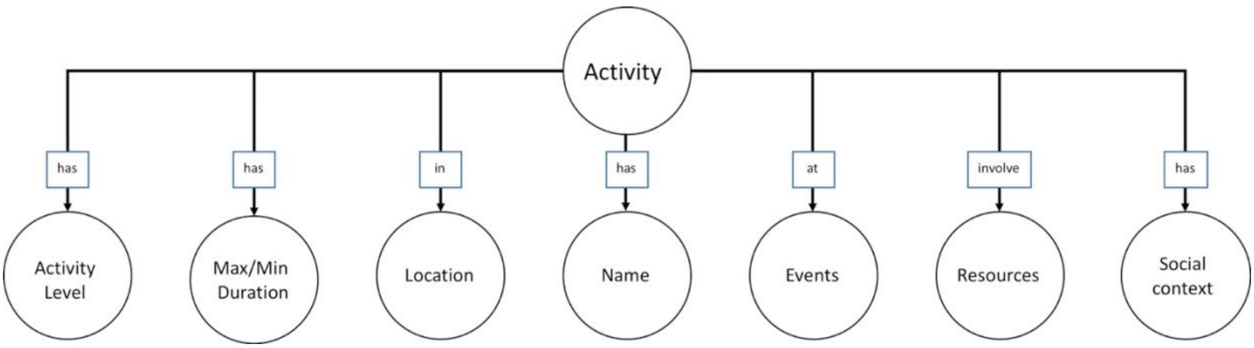

Fig. 2. Main concepts of proposed STRETCH Ontology.

3.3 Events detection rules

Rules are used to derive any events that describe the change in the smart home environment. The rules rely on the pre-processed sensor data. The rules are designed to detect eleven events. Table 1 shows the events and the related sensors for each event. The proposed rules are designed to detect the change in sensor events, which reflect a change in the environment. The rest of this section discusses the rule-based approach to detect events. Some algebraic notations are used to describe the rules, such as, ∧ for AND, ∨ for OR, Not for negation as failure, a comma for conjunction, and => for inference.

Table 1. List of targeted events and related sensors.

Event	Related Sensor
Room occupancy	PIR
Switch on a device	Energy
Switch off a device	Energy
Fridge door is opened	Contact switch
Fridge door is closed	Contact switch
Active	Wearable
Resting	Wearable
Steam on	Humidity
Steam off	Humidity
On bed	Pressure
Off bed	Pressure

- *Room occupancy*

The PIR sensor data is used for indoor localisation, where PIR generates the value 1 when motion is detected and 0 otherwise.

Threshold time windows:

Occ_window1: Short time when the participant gets out and comes back to the room.

Occ_window2: The time needed to move from one room to another.

PIR(Room, Current_value, previous_value, time)

PIR(Room,1,0, t_1) => assert (PIRon(Room, t_1))

PIR(Room,0,1, t_1) => assert (PIRoff(Room, t_1))

PIRon(Room, t_1)) ⋀ PIRoff(Room, t_2), $t_1 < t_2$ => assert(Occupied(Room, t_1, t_2))

If the PIR deactivated for a short time and then activated again, that means the older person left the room for a very short time and returned. The room occupation is thus extended. The following rule explains that:

Occupied(Room, t1, t2) ⋀ PIRoff(Room, t3) ⋀PIRon(Room, t4) ⋀ t1<t3<t4<t2 ⋀ t4-t3< Occ_window1 => assert(Modify(Occupied(Room, t1, t4))

If PIR sensor is activated in more than two rooms with in time window more than that is needed to move from one room to another, then more than one person is at the home. Recognising multiple occupancy is important for older people's care to recognise social interaction and get more information about who performs activities in the home.

Occupied(Room1, t_1,t_2) ⋀ Occupied(Room2, t_3,t_4) ⋀ $t_1<t_3<t_2<t_4$ ⋀ t_2-t_3> Occ_window2 => assert(Multi occupancy(t3,t4)

Recognising faulty PIR sensor can be performed through the following rule.

PIRoff(Room1, t_1) ⋀NOT(PIRon(Room2, t_2)) ⋀ t1- t2 > Occ_window1 => assert(FaultyPIR(Room2)).

- *Steam on/ off*

The humidity sensor reports the percent level of humidity in in the room. Increasing the reading above a certain threshold means that there is an activity that involves generating steam, such as preparing a meal or a drink or having a shower.

Threshold values:

Baseline_humidity: The humidity when there is no source of steam.

Humidity_sensor(room, current_reading, previous_reading, time)

Humidity_sensor(room1, >=X Baseline_humidity, <= Baseline_humidity, t_1)=> assert(Steam_ON(room1, t_1)).

Humidity_sensor(room1, <= Baseline_humidity, >=X Baseline_humidity, t_2)=> assert(Steam_OFF(room1, t_2)).

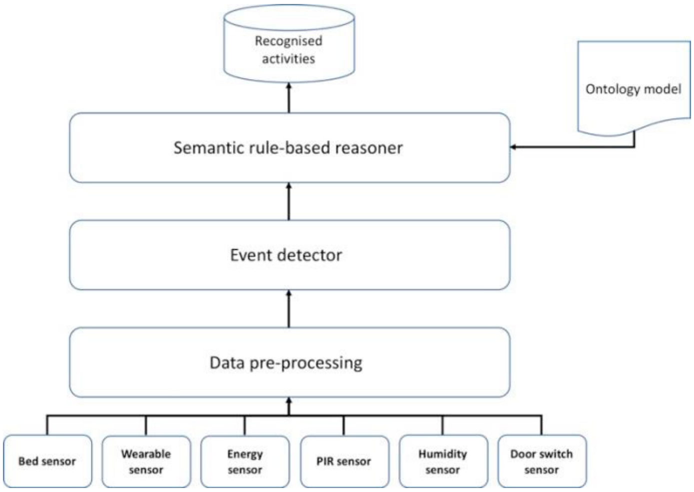

Fig. 3. Activity recognition framework architecture.

- *Switch on/off a device*

The STRETCH sensor platform is fitted with an energy sensor, which reports the electricity combustion. A sensor reading can be used to detect the event of switching an electric device on or off.

Threshold values:

Ideal_reading: the reading when none of the monitoring devices are working.

Reading_when_device_on: The consumption of device + Ideal_reading.

Energy_sensor (current_reading, previous_reading, time)

Energy_sensor (>= Reading_when_device_on, Ideal_reading, t_1*) => assert (SwitchON(device,* t_1*))*

Energy_sensor (Ideal_reading, >= Reading_when_device_on, t_2*) => assert (SwitchOFF(device,* t_2*))*

- *Fridge_door is opened/ closed*

The door contact sensor reports value 0 if the door open and 1 when it is shut. The refrigerator access event is triggered when the door stays open for more than (Door_window) seconds.

Threshold values:

Door_window: The minimum time needed to take food from the fridge.

Door (Room/ fridge, current value, previous value, time)

Door (fridge, 0,1, t_1*) => assert(Door _opened(fridge,* t_1*)).*

Door (fridge, 1,0, t_2*) => assert(Door _closed(fridge,* t_2*)).*

Door_opened(fridge, t_1*) Λ Door_closed(fridge, t2) Λ* t_2*-* t_1*>= Door_window => Fridge_usage (*t_1*,* t_2*)*

- *Active/ Resting*

The wearable registers acceleration in three directions (X, Y, Z). The participant is considered active if the magnitude of acceleration is above a certain threshold. The accelerometer produces 25 readings per second. The average magnitude of the 25 readings will be used to detect the level of activity. The average magnitude should be above a threshold value of more than a specific window of time to be considered as activity.

$$\mathrm{AVR\_ACC} = \frac{\sum_{i=1}^{25}\sqrt{x_i^2 + y_i^2 + z_i^2}}{25}$$

Threshold values:

Accelerometer_threshold: The value of acceleration that can be considered as movement.

Activity_window: The minimum time can be considered as a duration of activity.

Wearable (current_ wearable_AVR_ACC, previous_ wearable_ AVR_ACC, time)
Wearable (>=Accelerometer_threshold, < Accelerometer_threshold, t_1) =>assert(action(t_1)).
Wearable (<Accelerometer_threshold, >= Accelerometer_threshold, t_2) =>assert(rest(t_2)).
action(t_1) Λ rest(t_2) Λ(t_2- t_1)>= Activity_window =>assert (active(t_2- t_1)).
rest(t_3) Λ action(t_4) Λ (t_4- t_3)>= Activity_window =>assert (Resting(t_3- t_4)).

- *Bed on/off*

The bed pressure sensor produces a value of 1 when the older adult is on the bed and 0 when no one is on the bed.
Bed_sensor(current_reading, previous_reading, time)
Bed_sensor(1, 0, t_1)=> assert(Bed_on (t_1)).
Bed _sensor(0,1, t_2)=> assert(Bed_off(t_2)).

3.4 Activity recognition reasoner rules

At this layer, the events from the previous layer are used to recognise more semantically complex activities. The time duration of each activity has been identified based on preset minimum time, which is referred in this section as: Activity_window.

- *Living room activities:*

Threshold Values:
Watch_TV_window: The minimum time is needed for watching TV.
Resting_window: The minimum time is needed for resting.
Watching TV:
This activity involves using TV, being in living room for more that specific time window.
SwitchON(TV, t_1) ⋀ SwitchOFF(TV, t_2) ⋀ Occupied(Living room, t_5, t_6) ⋀ t_5< t_1< t_2< t_6 ⋀ t_1- t_2 >= Watch_TV_window=> watching TV(t_1, t_2).
SwitchON(TV, t_1) ⋀ Occupied(Living room, t_5, t_6) ⋀ t_5< t_1 < t_6⋀ t_1- t_6 >= Watch_TV_window => modify(watching TV(t_1, t_6)).
SwitchON(TV, t_1) ⋀ SwitchOFF(TV, t_2) ⋀ Occupied(Living room, t_5, t_6) ⋀ t_1< t_5< t_2< t_6 ⋀ t_5- t_2 >= Watch_TV_window =>modify(watching TV(t_5, t2)).
Resting/ Reading:
For this activity wearable sensor events are used to detect with in the rule to recognise this actitivty.
Occupied (Living room, t_1, t_2) ⋀ NOT (SwitchON(TV, t_3)) ⋀ Resting (t_4, t_5) ⋀ t_1< t_3 < t_2 ⋀ t_1< t_4 < t_5 < t_2 ⋀ t_5- t_4 >= Resting_window => Resting TV(t_4, t_5).

- *Kitchen activities*

Prepare a Meal and a Drink:
Threshold values:
Meal_window: the minimum time needed to prepare a meal
Drink_window: the minimum time needed to prepare a hot drink
Prepare Meal:
Occupied(Kitchen, t_7,t_8) ⋀SwitchON(appliances, t_1) ⋀ SwitchOFF(appliances, t_2) ⋀ Fridge_usage (t_3,t_4) ⋀ t_7<t_1< t_8< t_2 ⋀ t_7< t_3< t_4< t_8 ⋀ t_8-t_7>= Meal_window => assert (preparing meal(t_1, t_2))
Prepare Drink:
Occupied(Kitchen,t_7,t_8) ⋀ SwitchON(appliances,t_1) ⋀ SwitchOFF(appliances,t_2) ⋀ Steam_ON(Kitchen,t_5) ⋀ Steam_OFF(Kitchen,t_6) ⋀ t_7< t_1< t_8 ⋀ t_1< t_5< t_2< t_6 ⋀ t_8-t_7>= Drink_window => assert (preparing drink(t_1,t_2))

- *Bathroom Activities*

Threshold Values:
Shower_window: the minimum time needed to have a shower

Hygiene_window: the minimum time needed for hygiene activity

Having a Shower:

Occupied(Bathroom, t_1, t_2) ∧ Steam_ON(Bathroom, t_5) ∧ Steam_OFF(Bathroom, t_6) ∧ $t_1 < t_5 < t_2 < t_6$ ∧ t_2-t_1>= Shower_window ==> assert(Having_shower(t1, t2)).

Hygiene:

Occupied(Bathroom, t1, t2) ∧ NOT(Steam_ON(Bathroom, t5)) ∧ t1< t5< t2 ∧ t2-t1>= Hygiene_window => assert(Hygiene (t_1, t_2)).

- *Bedroom Activities*

Threshold Values:

Sleep_window: the minimum time that can be considered as sleep.

Occupied(bedroom, t_1, t_2) ∧ On_bed(t_3)∧ Off_bed(t_4) ∧NOT(active(t_5, t_6)) ∧ $t_1 < t_3 < t_4 < t_2$ ∧ $t_3 < t_5 < t_6 < t_4$ ∧ t_4-t_3>= Sleep_window => assert(sleeping (t_3, t_4)).

A reasoning engine will be used to infer the logical rules. A large number of reasoning engines have been proposed, most of which work well with Java, and several of them are available as open source software. Java Expert System Shell (Jess) [31] is one of the most commonly used engines, it consists of a rule base and an execution engine. The applicability of Jess for this application will investigated. Jess can be run within the Protégé [32] framework which is the tool used to develop the ontology. The reason for choosing Jess is that both Jess and Protégé are developed using Java. Therefore, one Java virtual machine can be used to run both of them at the same time.

4 Conclusion and Future Work

In this paper, a knowledge-based ADL recognition architecture has been proposed, which detects activities in real-time. An ADL ontology is part of the architecture. The work includes rules for sensor events detection and complex activity recognition. The proposed architecture is designed to overcome the lack of annotated ADL data problems and the variation or complex activity performance between different individuals. Future work may evaluate this architecture and expand the rules to handle more activities, to use the detected activities and events to study human behaviour and anomaly detection, and to tune and evolve the ontology for older people ADL recognition so that anomalies can be recognized early and interventions made before problems become serious. Finally, this architecture may be used as a base for developing active learning systems for data annotation.

Acknowledgements

This research was part funded by UK EPSRC grants EP/P01013X/1 (STRETCH) and EP/R013144/1 (SAUSE) and ERC grant 291652 (ASAP).

References

1] United Nations. World Population Ageing, 2017.

[2] Goonawardene, N., Toh, X. and Tan, H.P., (2017) "Sensor-driven detection of social isolation in community-dwelling elderly." *In International Conference on Human Aspects of IT for the Aged Population (pp. 378-392)*. Springer, Cham.

[3] Bersch, S., Azzi, D., Khusainov, R., Achumba, I. and Ries, J., (2014) "Sensor data acquisition and processing parameters for human activity classification." *Sensors*, 14(3), pp.4239-4270.

[4] Sanchez, V., Pfeiffer, C. and Skeie, N.O., (2017) "A review of smart house analysis methods for assisting older people living alone". *Journal of Sensor and Actuator Networks*, 6(3), p.11.

[5] Williams, A., Ganesan, D. and Hanson, A., (2007) "September. Aging in place: fall detection and localization in a distributed smart camera network". *In Proceedings of the 15th ACM international conference on Multimedia (pp. 892-901)*.

[6] Mozer, M.C., (1998) "The neural network house: An environment that adapts to its inhabitants." *In Proc. AAAI Spring Symp. Intelligent Environments* (Vol. 58).

[7] Bao, L. and Intille, S.S., (2004) "Activity recognition from user-annotated acceleration data." *In International conference on pervasive computing* (pp. 1-17). Springer, Berlin, Heidelberg.

[8 Patterson, D.J., Liao, L., Fox, D. and Kautz, H., (2003) "Inferring high-level behavior from low-level sensors." *In International Conference on Ubiquitous Computing* (pp. 73-89). Springer, Berlin, Heidelberg.

[9] Sung, M., DeVaul, R., Jimenez, S., Gips, J. and Pentland, A., (2004) "October. Shiver motion and core body temperature classification for wearable soldier health monitoring systems". *In Eighth international symposium on wearable computers* (Vol. 1, pp. 192-193). IEEE.

[10] Wang, A., Chen, G., Yang, J., Zhao, S. and Chang, C.Y., (2016) "A comparative study on human activity recognition using inertial sensors in a smartphone." *IEEE Sensors Journal*, 16(11), pp.4566-4578.

[11] Sixsmith, A.J., (2000) "An evaluation of an intelligent home monitoring system." *Journal of telemedicine and telecare*, 6(2), pp.63-72.

[12] Weingaertner, T., Hassfeld, S. and Dillmann, R., (1997) "Human motion analysis: A review." *In Proceedings of the 1997 IEEE Workshop on Motion of Non-Rigid and Articulated Objects (NAM'97)* (p. 90). IEEE Computer Society

[13] Weinland, D., Ronfard, R. and Boyer, E., (2011) "A survey of vision-based methods for action representation, segmentation and recognition." *Computer vision and image understanding*, 115(2), pp.224-241

[14] Van Laerhoven, K. and Aidoo, K., (2001) "Teaching context to applications." *Personal and Ubiquitous Computing,* 5(1), pp.46-49.

[15] Chen, L., Hoey, J., Nugent, C.D., Cook, D.J. and Yu, Z., (2012) "Sensor-based activity recognition." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6), pp.790-808.

[16] Bennasar, M., McCormick, C., Price, B., Gooch, D., Stuart, A., Mehta, V., Clare, C., Bennaceur, A., Cohen, J., Bandara, A., Levine, M., and Nuseibeh, N., (2019), "A Sensor Platform for Non-Invasive Remote Monitoring of Older Adults in Real Time." *In Proceedings of the 7th KES International Conference on Innovation in Medicine and Healthcare (KES-InMed-19)*.

[17] Civitarese, G., Bettini, C., Sztyler, T., Riboni, D. and Stuckenschmidt, H., (2018) "March. NECTAR: Knowledge-based collaborative active learning for activity recognition." *In 2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)* (pp. 1-10). IEEE.

[18] Woznowski, P., Tonkin, E. and Flach, P., (2018) "Activities of Daily Living Ontology for Ubiquitous Systems: Development and Evaluation." *Sensors*, 18(7), p.2361.

[19] Riboni, D., Sztyler, T., Civitarese, G. and Stuckenschmidt, H., (2016) "September. Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning." *In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (pp. 1-12). ACM.

[20] Loke, S.W., 2004. Representing and reasoning with situations for context-aware pervasive computing: a logic programming perspective. The Knowledge Engineering Review, 19(3), pp.213-233.

[21] Riboni, D. and Bettini, C., 2011. OWL 2 modeling and reasoning with complex human activities. Pervasive and Mobile Computing, 7(3), pp.379-395.

[22] Tapia, E.M., Choudhury, T. and Philipose, M., (2006) "May. Building reliable activity models using hierarchical shrinkage and mined ontology." *In International Conference on Pervasive Computing* (pp. 17-32). Springer, Berlin, Heidelberg.

[23] Wyatt, D., Philipose, M. and Choudhury, T., (2005) "Unsupervised activity recognition using automatically mined common sense." *In AAAI* (Vol. 5, pp. 21-27).

[24] Bouchard, B., Giroux, S. and Bouzouane, A., (2006) "A smart home agent for plan recognition of cognitively-impaired patients." J*CP*, 1(5), pp.53-62.

[25] Gruber, T.R., (1993) "A translation approach to portable ontology specifications." *Knowledge acquisition*, 5(2), pp.199-220.

[26] Chen, L., Nugent, C.D. and Wang, H., (2012) "A knowledge-driven approach to activity recognition in smart homes." *IEEE Transactions on Knowledge and Data Engineering*, 24(6), pp.961-974.

[27] Riboni, D. and Bettini, C., (2011) "OWL 2 modeling and reasoning with complex human activities." *Pervasive and Mobile Computing*, 7(3), pp.379-395.

[28] Salguero, A.G., Medina, J., Delatorre, P. and Espinilla, M., (2018) "Methodology for improving classification accuracy using ontologies: application in the recognition of activities of daily living." *Journal of Ambient Intelligence and Humanized Computing*, pp.1-18.

[29] Bae, I.H., (2014) "An ontology-based approach to ADL recognition in smart homes." *Future Generation Computer Systems*, 33, pp.32-41.

United Nations. World Population Ageing, 2017.

[30] Culmone, R., Giuliodori, P. and Quadrini, M., (2015) "Human activity recognition using a semantic ontology-based framework." *International Journal on Advances in Intelligent Systems*, 8(1, 2), pp.159-168.

[31] Horridge, M., Knublauch, H., Rector, A., Stevens, R. and Wroe, C., (2004) "A Practical Guide to Building OWL Ontologies Using the Protégé-OWL Plugin and CO-ODE Tools Edition 1.0." University of Manchester.

[32] Musen, M.A. (2015) "The Protégé project: A look back and a look forward." AI Matters. Association of Computing Machinery Specific Interest Group in Artificial Intelligence, 1(4).